

Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh

Development of an Automatic Flood Monitoring Device Using a Proximity Sensor and ESP32 with Remote Notification Features

Aditya Maulana Nst¹, Iwan Fitrianto Rahmad²

^aProgram Studi Rekayasa Sistem Komputer Universitas Potensi Utama, Jl.K.L Yos Sudarso KM 6.5 Tj.Mulia, Medan, 20241, Indonesia

^bProgram Studi Rekayasa Sistem Komputer Universitas Potensi Utama, Jl.K.L Yos Sudarso KM 6.5 Tj.Mulia, Medan, 20241, Indonesia
E-mail: adit24052003@gmail.com¹, iwanfitrah@yahoo.com²

ABSTRAK

Banjir merupakan salah satu bencana alam yang sering terjadi di berbagai wilayah Indonesia dan menimbulkan kerugian besar baik secara material maupun non-material. Minimnya sistem pemantauan dini menjadi salah satu penyebab keterlambatan dalam penanggulangan banjir. Penelitian ini bertujuan untuk mengembangkan alat pemantau banjir otomatis yang mampu mendeteksi ketinggian air secara real-time menggunakan sensor jarak berbasis ultrasonik yang terhubung dengan mikrokontroler ESP32. Alat ini dirancang agar dapat mengirimkan notifikasi jarak jauh secara otomatis melalui jaringan internet ketika ketinggian air melebihi ambang batas yang telah ditentukan. Metode pengembangan yang digunakan meliputi perancangan perangkat keras, pemrograman perangkat lunak, serta pengujian sistem secara langsung pada simulasi kondisi banjir. Hasil pengujian menunjukkan bahwa alat dapat bekerja secara efektif dalam memantau ketinggian air dengan tingkat akurasi tinggi dan memberikan peringatan secara cepat melalui pesan notifikasi. Dengan demikian, alat ini diharapkan dapat menjadi sistem peringatan dini yang membantu masyarakat dan pihak berwenang dalam mengantisipasi terjadinya banjir.

Kata Kunci : Banjir, Sensor Jarak, ESP32, Internet of Things (IoT), Notifikasi Jarak Jauh.

ABSTRACT

Floods are among the most frequent natural disasters in Indonesia, often causing significant material and non-material losses. The lack of early monitoring systems is one of the main factors contributing to delays in flood mitigation efforts. This study aims to develop an automatic flood monitoring device capable of detecting water levels in real time using an ultrasonic distance sensor integrated with an ESP32 microcontroller. The device is designed to send remote notifications automatically via the internet when the water level exceeds a predetermined threshold. The development process includes hardware design, software programming, and system testing under simulated flood conditions. The test results show that the device can effectively monitor water levels with high accuracy and provide timely warnings through notification messages. Therefore, this tool is expected to serve as an early warning system to help communities and authorities anticipate potential flooding events.

Keywords : Flood, Distance Sensor, ESP32, Internet of Things (IoT), Remote Notification.

Info Artikel:

Disubmit:

Direview:

Diterima :

Copyright © 2025 - Senadimu. All rights reserved.

1. PENDAHULUAN

Banjir merupakan salah satu bencana alam yang sering terjadi di berbagai wilayah Indonesia, terutama saat musim hujan. Dampak dari banjir tidak hanya merusak infrastruktur dan mengganggu aktivitas masyarakat, tetapi juga dapat mengancam keselamatan jiwa. Salah satu faktor yang memperparah dampak banjir adalah keterlambatan informasi kepada masyarakat terkait naiknya permukaan air di suatu wilayah. Oleh karena itu, sistem peringatan dini (early warning system) yang cepat, akurat, dan dapat diakses secara luas sangat diperlukan [1].

Seiring perkembangan teknologi, berbagai perangkat mikrokontroler dan sensor telah memungkinkan pengembangan alat pemantau kondisi lingkungan secara otomatis dan real-time. ESP32, sebagai salah satu mikrokontroler yang memiliki kemampuan koneksi Wi-Fi, sangat potensial digunakan dalam sistem pemantauan berbasis Internet of Things (IoT). Dengan memanfaatkan sensor jarak seperti ultrasonik (misalnya HC-SR04), alat ini dapat mengukur ketinggian air secara terus menerus dan mengirimkan data tersebut ke platform digital.

Internet of Things (IoT) merupakan jaringan berbagai macam sensor ataupun penginderaan yang saling melakukan komunikasi pengolahan data atau informasi, dengan komunikasi yang terdiri dari global positioning system (GPS), laser scanner, sensor infra merah, radio frekuensi identification (RFID) ataupun sistem lainnya dan internet [2]

Untuk memastikan informasi terkait potensi banjir dapat diterima oleh masyarakat secara cepat, diperlukan integrasi sistem notifikasi jarak jauh. Turrit Telegram, sebagai salah satu aplikasi pesan instan yang banyak digunakan, menyediakan API (*Application Programming Interface*) yang memungkinkan pengiriman pesan otomatis melalui bot. Dengan demikian, ketika alat mendeteksi bahwa ketinggian air telah melewati batas tertentu, sistem dapat langsung mengirimkan notifikasi ke pengguna melalui Turrit Telegram secara *real-time*.

Berdasarkan uraian tersebut, maka perlu dikembangkan sebuah alat pemantau banjir otomatis yang tidak hanya dapat mengukur ketinggian air, tetapi juga mampu mengirimkan peringatan dini kepada pengguna secara langsung melalui Turrit Telegram. Dengan adanya alat ini, diharapkan masyarakat dapat lebih waspada dan dapat mengambil tindakan preventif lebih awal untuk mengurangi dampak banjir.

Berdasarkan penelitian yang dilakukan oleh [3] Untuk itu penulis mengembangkan alat pendeteksi dan monitoring banjir menggunakan ESP32 berbasis IoT, untuk memberikan informasi real-time ketinggian air dan memberikan peringatan melalui server MQTT pada notifikasi aplikasi mobile agar wilayah tersebut dapat bertindak untuk mengevakuasi dirinya sebelum terjadinya banjir. Penulis menggunakan metode penelitian berupa pengumpulan data dan observasi yang memperoleh hasil pengujian bahwa Arduino IDE mempublish data kepada MQTT Box dan dapat dihubungkan ke aplikasi kodular [4]. Alat pendeteksi banjir berhasil diuji coba dan alat mampu bekerja sesuai perintah [5].

Hasil akhir dari penelitian ini menunjukkan bahwa dengan penempatan komponen yang tepat serta spesifikasi dan pemrograman yang disesuaikan dengan ukuran miniatur bendungan, Pengujian sampel pada setiap komponen sistem menghasilkan nilai tingkat akurasi sangat baik, gelombang air sangat mempengaruhi sistem terlebih lagi pada saat pendeteksian ketinggian air [6]. Analisa data yang didapatkan pada pengujian sistem didapatkan hasil kinerja yang berjalan dengan baik [7]. Banjir sering terjadi, terutama di wilayah barat Indonesia, yang mengalami curah hujan lebih tinggi dibandingkan dengan daerah tengah dan timur [8]. Kajian literatur ini bertujuan untuk memberi informasi serta mengetahui tentang cara mengembangkan sistem peringatan dini banjir yang berbasis Internet of Things (IoT), menggunakan sensor ultrasonik HC-SR04 dan mikrokontroler NodeMCU ESP8266 [9]. Sistem ini dirancang untuk memantau ketinggian air secara real-time, menyimpan data di platform ThingSpeak, serta mengirimkan peringatan melalui aplikasi Telegram. Dengan mengelompokkan ketinggian air ke dalam empat kategori—aman, siaga 1, siaga 2, dan bahaya—sistem ini memberikan peringatan melalui buzzer dan notifikasi kepada masyarakat [10].

Data lingkungan yang dikumpulkan diolah oleh mikrokontroler ESP32, kemudian disimpan dan dikelola melalui Firebase untuk memfasilitasi akses real-time oleh pengguna. Penelitian ini juga mengevaluasi kinerja sistem dalam berbagai kondisi lingkungan [11]. Hasil pengujian menunjukkan akurasi tinggi pada deteksi ketinggian air hingga 179 cm, serta kemampuan sistem dalam memberikan peringatan dini secara cepat dan efisien melalui aplikasi berbasis Kodular [12]. Penggunaan sistem ini diharapkan dapat mengurangi dampak buruk banjir, meningkatkan kesiapsiagaan masyarakat, dan mendukung pengambilan keputusan yang lebih baik di tingkat lokal dan regional [13]. Data yang terkumpul dikirim secara langsung ke server cloud melalui koneksi IoT. Melalui antarmuka blynk, pengguna dapat memantau kondisi banjir secara real-time [14]. Pengujian prototipe menunjukkan keberhasilan dalam mendeteksi tinggi air dengan akurat. Solusi ini diharapkan dapat memberikan informasi dini kepada masyarakat dan pihak berwenang untuk mengambil tindakan preventif yang tepat dalam menghadapi potensi bencana banjir [15].

2. METODE

2.1. Metode Pengumpulan Data

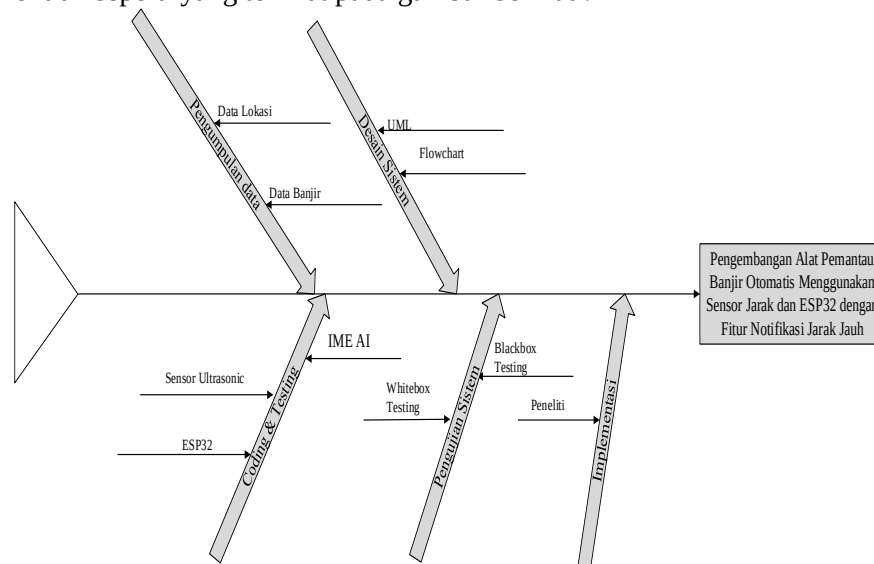
Didalam proses perancangan aplikasi ini diterapkan metode penelitian yang mendukung didalam proses perancangan, diantaranya adalah :

A. Studi kepustakaan (*Library Research*)

Melakukan kegiatan studi pustaka baik dari media internet, jurnal, buku –buku yang berhubungan dengan kegiatan perancangan sebuah aplikasi, seperti dasar – dasar membangun sebuah Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh.

B. Metode Perancangan Aplikasi

Adapun prosedur pembangunan aplikasi yang saya lakukan adalah menggunakan metode rancangan penelitian seperti yang terlihat pada gambar berikut :



Gambar 1. Rancangan Alur Fishbone

Adapun keterangan dari *fishbone* penelitian adalah sebagai berikut :

a. Pengumpulan data

Data dikumpulkan dengan cara melakukan observasi secara langsung pada lingkungan rumah dan melakukan studi literatur dari berbagai referensi jurnal. Data yang diperoleh berupa analisis kebutuhan terkait hardware dan software yang digunakan untuk perancangan sistem, seperti Arduino IDE dan Fritzing. Pada tahap ini penulis melakukan analisa sehingga dapat di definisikan kebutuhan-kebutuhan sistem meliputi input, output, operasi dan resources sehingga dapat terbentuk suatu alat yang baru yang lebih handal (Rahmad et al., 2021)

b. Desain sistem

Pembuatan prototipe diawali dengan membuat skema diagram pada Fritzing dan membuat desain Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh. Desain tersebut akan dicetak menggunakan bahan Akrilik yang telah disesuaikan dengan komponen yang digunakan

c. Coding dan testing

Pengkodean sistem dilakukan menggunakan Arduino IDE. Pada tahap ini, prototipe yang telah jadi akan ditanamkan program agar dapat bekerja sesuai alur flowchart.

d. Pengujian Sistem

Program yang telah dibuat akan dilakukan pengujian terlebih dahulu apakah terdapat error atau tidak. Jika tidak maka akan dilanjutkan ke tahap evaluasi.

e. Implementasi

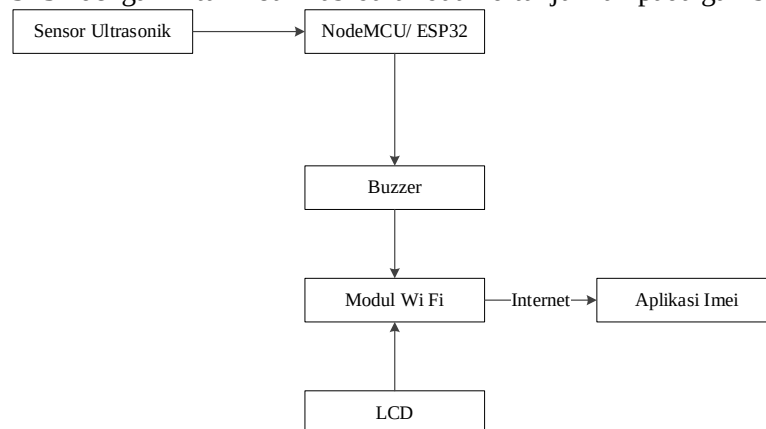
Tahap evaluasi sistem berguna untuk mengetahui apakah terdapat kendala saat sistem tersebut digunakan. Pada tahap ini akan dilakukan pengujian terhadap kinerja sensor, arah gerak dan kemampuan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh.

3. HASIL DAN PEMBAHASAN

3.1. Pembahasan

Banjir merupakan salah satu bencana alam yang sering terjadi di berbagai wilayah Indonesia, terutama saat musim hujan. Dampak dari banjir tidak hanya merusak infrastruktur dan mengganggu aktivitas masyarakat, tetapi juga dapat mengancam keselamatan jiwa. Salah satu faktor yang memperparah dampak banjir adalah keterlambatan informasi kepada masyarakat terkait naiknya permukaan air di suatu wilayah. Oleh karena itu, sistem peringatan dini (early warning system) yang cepat, akurat, dan dapat diakses secara luas sangat diperlukan [3]. Seiring perkembangan teknologi, berbagai perangkat mikrokontroler dan sensor telah memungkinkan pengembangan alat pemantau kondisi lingkungan secara otomatis dan real-time. ESP32, sebagai salah satu mikrokontroler yang memiliki kemampuan koneksi Wi-Fi, sangat potensial digunakan dalam sistem pemantauan berbasis Internet of Things (IoT). Dengan memanfaatkan sensor jarak seperti ultrasonic (misalnya HC-SR04), alat ini dapat mengukur ketinggian air secara terus menerus dan mengirimkan data tersebut ke platform digital.

Secara garis besar, membuat Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh ditunjukkan pada gambar berikut :



Gambar 2. Diagram Blok Rangkaian

Berikut adalah penjelasan dari diagram blok rangkaian diatas :

1. Deteksi Ketinggian Air oleh Sensor Ultrasonik

Tahap pertama dimulai dari sensor ultrasonik yang berfungsi untuk mengukur jarak antara sensor dan permukaan air. Sensor ini memancarkan gelombang ultrasonik dan menghitung waktu pantulan gelombang tersebut. Semakin tinggi permukaan air, maka jarak yang terdeteksi semakin kecil. Data hasil pengukuran inilah yang menjadi indikator utama dalam menentukan tingkat ketinggian air pada suatu area pemantauan banjir.

2. Pemrosesan Data oleh NodeMCU/ESP32

Data jarak yang diperoleh dari sensor ultrasonik kemudian dikirim ke mikrokontroler NodeMCU atau ESP32. Komponen ini berfungsi sebagai otak sistem yang memproses data sensor dan melakukan analisis sederhana. ESP32 membandingkan hasil pembacaan dengan batas ambang yang telah ditentukan (misalnya jarak ≤ 30 cm dianggap berbahaya). Jika data menunjukkan kondisi siaga atau bahaya, maka ESP32 akan mengaktifkan sistem peringatan dan komunikasi.

3. Aktivasi Sistem Peringatan Lokal (Buzzer)

Ketika NodeMCU/ESP32 mendeteksi ketinggian air yang melebihi batas aman, sistem akan mengaktifkan buzzer sebagai tanda peringatan lokal. Bunyi buzzer ini berfungsi untuk memberi peringatan langsung kepada masyarakat atau petugas di sekitar lokasi agar segera melakukan tindakan antisipasi terhadap potensi banjir yang terjadi.

4. Pengiriman Data ke Modul Wi-Fi dan Internet

Setelah alarm lokal aktif, ESP32 juga akan mengirimkan data hasil deteksi ke modul Wi-Fi yang sudah terintegrasi dalam perangkat. Melalui koneksi internet, informasi tersebut kemudian dikirim ke aplikasi pemantauan jarak jauh, seperti aplikasi berbasis web atau platform notifikasi (misalnya IFTTT atau Telegram). Dengan demikian, kondisi tinggi air dapat dipantau dari mana saja secara real-time.

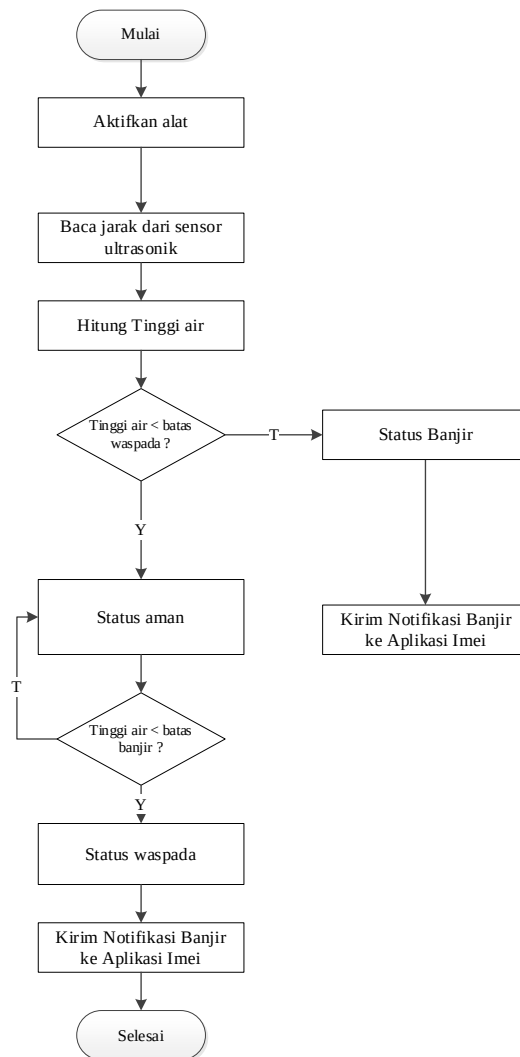
5. Penampilan Data pada LCD

Selain mengirimkan data ke internet, ESP32 juga menampilkan hasil pembacaan jarak air pada layar LCD. Tampilan LCD memberikan informasi langsung di lokasi seperti nilai jarak air dalam satuan sentimeter dan status kondisi (aman, siaga, atau bahaya). Fitur ini membantu pengguna yang berada di lokasi alat untuk mengetahui kondisi tanpa harus membuka aplikasi.

6. Notifikasi Jarak Jauh melalui Aplikasi IMEI / Internet

Tahap terakhir adalah pengiriman notifikasi jarak jauh melalui aplikasi yang terhubung ke internet. Data dari modul Wi-Fi diteruskan ke aplikasi pemantauan (IMEI app atau server cloud) sehingga pengguna akan menerima pemberitahuan otomatis di ponsel ketika ketinggian air sudah mencapai level berbahaya. Dengan sistem ini, peringatan dini banjir dapat diterima lebih cepat, akurat, dan tidak terbatas jarak.

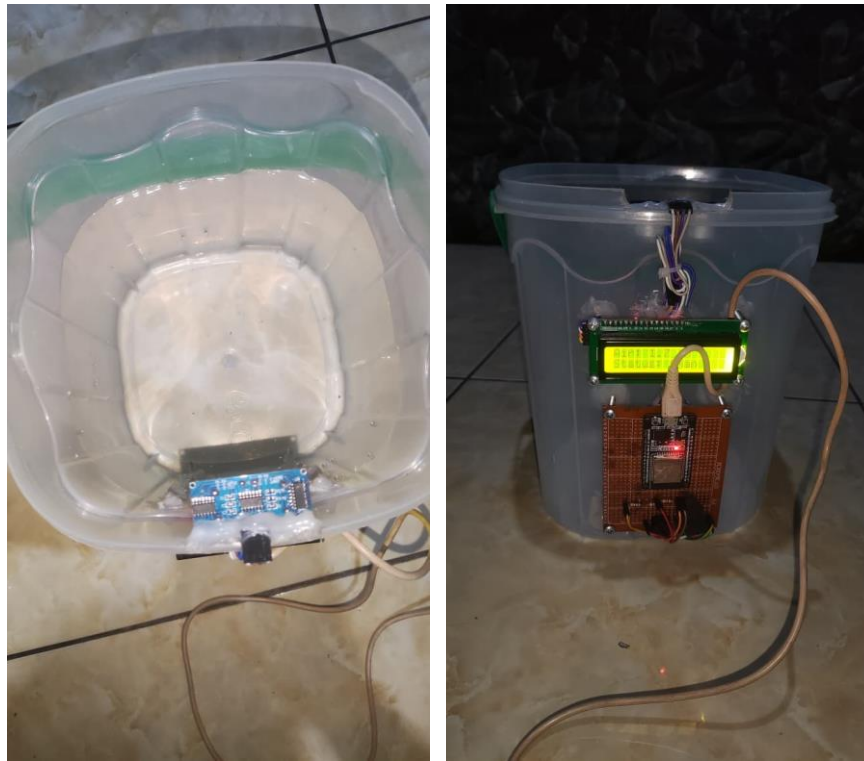
Berikut adalah *flowchart* dari jalannya perangkat :

Gambar 3. *Flowchart* Perangkat

3.2. Hasil

1. Tampilan Rangkaian Alat

Tampilan miniatur Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh yang akan dibuat merupakan skema dari jaringan yang akan dibuat IoT ditunjukkan pada gambar 4

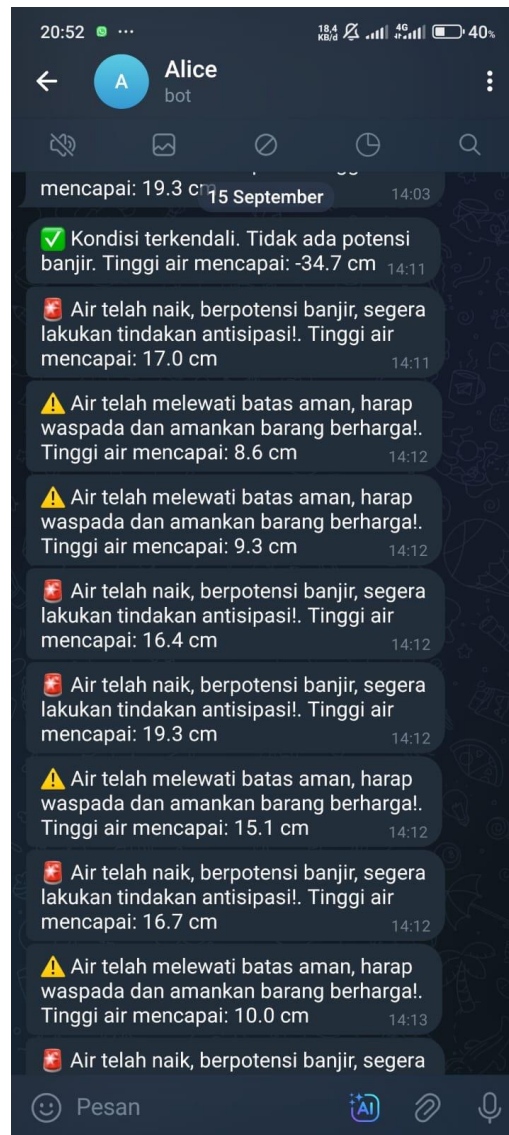


Gambar 4. Rangkaian Alat

Gambar diatas menunjukkan sebuah prototipe alat pemantau banjir otomatis yang dirancang menggunakan wadah plastik sebagai penampung simulasi air, dilengkapi dengan sensor dan komponen elektronik. Pada bagian atas wadah, terpasang sensor ultrasonik yang berfungsi untuk mengukur jarak permukaan air terhadap sensor, sehingga dapat menghitung ketinggian air. Hasil pengukuran ini dikirim ke mikrokontroler ESP32 yang terpasang pada papan PCB di bagian depan wadah. ESP32 bertugas memproses data, menentukan status ketinggian air (aman, waspada, atau banjir), dan menampilkan informasi tersebut pada LCD 16x2 yang juga terpasang di bagian depan. LCD menampilkan data berupa tinggi air dalam sentimeter dan status kondisi saat ini. Selain itu, rangkaian ini juga terhubung ke jaringan melalui modul Wi-Fi bawaan ESP32 untuk mengirimkan notifikasi ke aplikasi IMEI AI Messenger, terutama ketika status berubah menjadi waspada atau banjir. Sistem ini juga dilengkapi buzzer yang memberikan peringatan suara dengan pola berbeda sesuai kondisi air, sehingga pengguna dapat memperoleh peringatan baik secara lokal maupun jarak jauh.

2. Tampilan Aplikasi

Tampilan aplikasi Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh yang akan dibuat merupakan skema dari jaringan yang akan dibuat IoT ditunjukkan pada gambar 5 :



Gambar 5. Tampilan Notifikasi IME

Gambar tersebut menunjukkan tampilan notifikasi yang dikirimkan oleh sistem pemantau banjir otomatis melalui aplikasi IME Messenger. Notifikasi berisi dua jenis peringatan utama:

1. Status WASPADA ditandai dengan ikon peringatan  dan informasi tinggi air (misalnya "*Tinggi air: 9.3 cm*"). Status ini muncul ketika tinggi air berada di ambang batas awal yang berpotensi naik menjadi banjir, sehingga pengguna mendapat peringatan untuk bersiap.
2. BANJIR — ditandai juga dengan ikon peringatan  tetapi disertai kata "BANJIR!" di awal pesan, diikuti keterangan tinggi air (misalnya "*Tinggi air: 19.3 cm*"). Status ini menandakan tinggi air sudah melewati batas aman, sehingga diperlukan tindakan segera.

Sistem mengirimkan pesan ini secara real-time setiap kali sensor mendeteksi perubahan status air. Notifikasi bergantian antara Waspada dan Banjir sesuai fluktuasi tinggi air yang terukur. Hal ini memungkinkan pengguna memantau perkembangan kondisi air dari jarak jauh, tanpa harus berada di lokasi alat.

3. Tampilan Coding Alat

Tampilan pada IoT (*Internet Of Things*) dengan menggunakan arduino merupakan proses awal untuk membuat Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh yang digunakan menggunakan wifi handphone yang yang digunakan, adapun langkah – langkahnya sebagai berikut :

1. Tampilan Coding Alat

Berikut ini merupakan coding alat Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh menggunakan internet:

```
#include <WiFi.h>
#include <HTTPClient.h>
#include <LiquidCrystal_I2C.h>

// =====
// Konfigurasi WiFi
// =====
const char* ssid = "HP POCO F4";
const char* password = "Indonesia";

// =====
// Konfigurasi Telegram Bot
// =====
String botToken = "8372714221:AAE-R3ZIIu51osE_UBGPoretzjXSwh0L0TY";
String chatID = "7991051723";

// =====
// Konfigurasi Pin Sensor & Aktuator
// =====
#define PIN_TRIG 5
#define PIN_ECHO 18
#define PIN_BUZZER 19

// =====
// LCD I2C 16x2
// =====
LiquidCrystal_I2C lcd(0x27, 16, 2);

// =====
// Variabel Logika

const unsigned long intervalAman = 300000; // 5 menit
const unsigned long intervalWaspada = 30000; // 30 detik
const unsigned long intervalBanjir = 5000; // 5 detik

// Untuk buzzer waspada
const unsigned long intervalBeepOn = 300; // durasi beep
const unsigned long intervalBeepOff = 300; // durasi jeda beep
bool isBeepOn = false;

// =====
// Fungsi: Kirim Pesan Telegram
// =====
void kirimPesanTelegram(String pesan) {
  if (WiFi.status() == WL_CONNECTED) {
    HTTPClient http;
    String url = "https://api.telegram.org/bot" + botToken +
"/sendMessage?chat_id=" + chatID + "&text=" + pesan;
    http.begin(url);
    int httpCode = http.GET();
    if (httpCode > 0) {
      Serial.println("Pesan terkirim: " + pesan);
    } else {
      Serial.println("Gagal kirim pesan");
    }
    http.end();
  } else {
    Serial.println("WiFi tidak terkoneksi");
  }
}
```

```

lcd.backlight();

pinMode(PIN_TRIG, OUTPUT);
pinMode(PIN_ECHO, INPUT);
pinMode(PIN_BUZZER, OUTPUT);

jarakSensor = bacaJarak();
tinggiAir = tinggiMaks - jarakSensor;

if (tinggiAir < batasAman) {
  statusAir = "Aman";
} else if (tinggiAir >= batasAman && tinggiAir < batasWaspada) {
  statusAir = "Waspada";
} else {
  statusAir = "Banjir";
}

statusSebelumnya = statusAir;
kirimPesanTelegram(getPesanStatus(statusAir, tinggiAir));
previousMillisKirim = millis();
}

// =====
// Fungsi: Baca Jarak dari Sensor Ultrasonik
// =====
float bacaJarak() {
  digitalWrite(PIN_TRIG, LOW);
  delayMicroseconds(2);

```

Gambar 6. Tampilan Coding Alat

Pengujian dilakukan dengan mensimulasikan kenaikan dan penurunan permukaan air pada wadah uji. Sensor ultrasonik membaca jarak air terhadap sensor, lalu sistem memprosesnya untuk menentukan status:

1. AMAN: Tinggi air di bawah batas waspada.
2. WASPADA: Tinggi air mencapai batas awal potensi banjir.
3. BANJIR: Tinggi air melewati batas aman yang ditentukan.

Pengujian mencakup:

1. Respon pembacaan sensor terhadap perubahan tinggi air.
2. Akurasi tampilan LCD dalam menampilkan tinggi air dan status.
3. Fungsi buzzer dalam membedakan bunyi antara status WASPADA dan BANJIR.
4. Pengiriman notifikasi ke IME secara real-time.
5. ketentuan nilai

AMAN: tinggi air < 8 cm → tidak ada notifikasi, buzzer mati.

WASPADA: 8–16cm → notifikasi Ya, buzzer beep pendek/berulang.

BANJIR: > 16.0 cm → notifikasi Ya, buzzer beep panjang/kontinu.

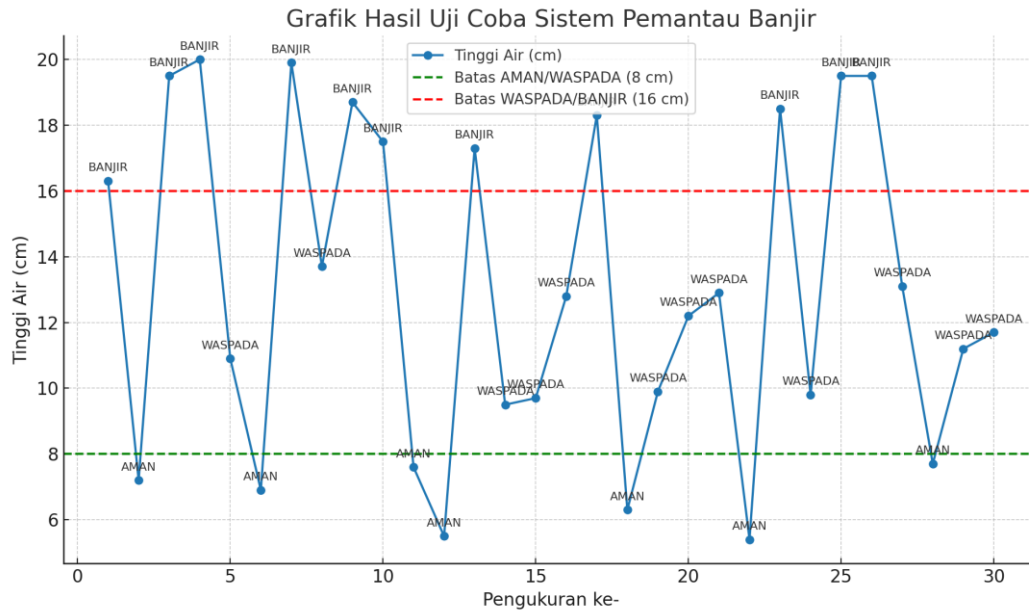
Dan berikut ini merupakan hasil uji coba dari sistem yang di bangun :

Tabel 1. Uji Coba Sistem

No	Tinggi Air (cm)	Status	Notifikasi	Buzzer
1	16,3	BANJIR	Ya	Beep panjang/kontinu
2	7,2	AMAN	Tidak	Mati
3	19,5	BANJIR	Ya	Beep panjang/kontinu
4	20	BANJIR	Ya	Beep panjang/kontinu
5	10,9	WASPADA	Ya	Beep pendek/berulang
6	6,9	AMAN	Tidak	Mati
7	19,9	BANJIR	Ya	Beep panjang/kontinu

No	Tinggi Air (cm)	Status	Notifikasi	Buzzer
8	13,7	WASPADA	Ya	Beep pendek/berulang
9	18,7	BANJIR	Ya	Beep panjang/kontinu
10	17,5	BANJIR	Ya	Beep panjang/kontinu
11	7,6	AMAN	Tidak	Mati
12	5,5	AMAN	Tidak	Mati
13	17,3	BANJIR	Ya	Beep panjang/kontinu
14	9,5	WASPADA	Ya	Beep pendek/berulang
15	9,7	WASPADA	Ya	Beep pendek/berulang
16	12,8	WASPADA	Ya	Beep pendek/berulang
17	18,3	BANJIR	Ya	Beep panjang/kontinu
18	6,3	AMAN	Tidak	Mati
19	9,9	WASPADA	Ya	Beep pendek/berulang
20	12,2	WASPADA	Ya	Beep pendek/berulang
21	12,9	WASPADA	Ya	Beep pendek/berulang
22	5,4	AMAN	Tidak	Mati
23	18,5	BANJIR	Ya	Beep panjang/kontinu
24	9,8	WASPADA	Ya	Beep pendek/berulang
25	19,5	BANJIR	Ya	Beep panjang/kontinu
26	19,5	BANJIR	Ya	Beep panjang/kontinu
27	13,1	WASPADA	Ya	Beep pendek/berulang
28	7,7	AMAN	Tidak	Mati
29	11,2	WASPADA	Ya	Beep pendek/berulang
30	11,7	WASPADA	Ya	Beep pendek/berulang

Berdasarkan pengujian, sistem berhasil bekerja sesuai rancangan maka berikut ini Adalah grafik hasil uji coba sistem :



Gambar 7. Grafik Uji Coba

4. KESIMPULAN

Berdasarkan perancangan, pembuatan, dan pengujian alat Pengembangan Alat Pemantau Banjir Otomatis Menggunakan Sensor Jarak dan ESP32 dengan Fitur Notifikasi Jarak Jauh menggunakan teknologi IoT, dapat diambil beberapa kesimpulan sebagai berikut:

1. Sistem pemantau banjir otomatis yang dibangun berhasil mendeteksi ketinggian air menggunakan sensor ultrasonik dengan akurasi yang memadai.
2. Fitur klasifikasi status banjir (AMAN, WASPADA, BANJIR) berfungsi sesuai dengan batas yang ditentukan, yaitu AMAN: Tinggi air < 8 cm, tidak ada notifikasi dan buzzer mati, WASPADA: Tinggi air 8–16 cm, mengirim notifikasi ke aplikasi IME, buzzer bunyi pendek berulang DAN BANJIR: Tinggi air > 16 cm, mengirim notifikasi ke aplikasi IME Messenger, buzzer bunyi panjang/kontinu.
3. Sistem mampu memberikan peringatan dini melalui IME Messenger sehingga pengguna dapat segera melakukan tindakan pencegahan.
4. Pengujian lapangan menunjukkan sistem berjalan stabil, dengan respon cepat pada perubahan tinggi air, dan informasi status ditampilkan secara real-time pada LCD.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada Universitas Potensi Utama yang telah banyak memberikan masukan dan saran dalam penyelesaian penelitian ini.

REFERENSI

- [1] Akmalano, R., & Santoso, D. B. (2025). *DETEKSI DINI BENCANA BANJIR DI PEMUKIMAN PERKOTAAN DENGAN TEKNOLOGI INTERNET OF THINGS (IOT)*. 9(2), 1867–1873.
- [2] Rahmad, I. F., Tanti, L., Triandi, B., Komputer, I., Komputer, R. S., & Utama, U. P. (2021). Wireless Sensor Network Sebagai Penentu Lokasi Kebakaran Hutan. *TIN: Terapan Informatika Nusantara*, 2(3), 138–144.

- [3] Anggasemara, N. L. K. I., Suarjaya, I. M. A. D., & Bayupati, I. P. A. (2023). Rancang Bangun Penyedot Debu Berbasis Internet of Things. *Walisongo Journal of Information Technology*, 5(1), 1–14. <https://doi.org/10.21580/wjit.2023.5.1.16123>
- [4] Anwar, N., Azis, W. A., Irawan, B., & Widodo, A. M. (2022). Perancangan Sistem Kendali Kunci Keamanan Pintu Gerbang Berbasis Arduino Dan QR-Code. In *Jurnal Algoritma, Logika dan Komputasi* (Vol. 5, Issue 1). <https://doi.org/10.30813/j-alu.v5i1.3598>
- [5] Dasril, D., Indou, H., & Suppa, R. (2024). Prototype Alat Pendeteksi Banjir Menggunakan Arduino Berbasis Iot. *Jurnal Informatika Dan Teknik Elektro Terapan*, 12(3). <https://doi.org/10.23960/jitet.v12i3.5135>
- [6] Energi, U. D. A. N. (2021). *Jurnal Pendidikan Fisika Tadulako Online Description of Understanding on Work and Energy Graphs of High School Students*. 9(August), 13–17.
- [7] Fahlevi, I., & Ardiansyah, M. (2022). Implementasi Alarm Kebakaran Otomatis Berbasis (IOT) Internet Of Things Menggunakan Metode Fuzzy Logic Pada Ruang Server Jaringan (Studi Kasus : YBY. Net). *Informatika*, 3(2), 2022.
- [8] Fragastia, V. A., Rahmad, I. F., Studi, P., Industri, T., Studi, P., Informatika, T., Utama, U. P., Deli, M., & Utara, S. (n.d.). *Penerapan Internet Of Things (IoT) Untuk Mendeteksi Kadar Alkohol Pada Pengendara Mobil Vidi Agung Fragastia (1) , Iwan Fitrianto Rahmad (2)*. 1, 11–19.
- [9] Mardhatillah, A., & S.Kom., M.Sc, D. D. (2024). Sistem Notifikasi dan Kontrolling Smart Home Berbasis Internet of Things. *INOVTEK Polbeng - Seri Informatika*, 9(1), 228–242. <https://doi.org/10.35314/isi.v9i1.4043>
- [10] Pramudya, R., & Laksmiati, D. (2023). Perancangan Sistem Kendali Irigasi Saluran Sekunder Dan Otomatisasi Pengumpulan Sampah Pada Aliran Air Sawah Menggunakan (Jurnal Sistem ..., 11. <https://ejournal.indobarunasional.ac.id/index.php/jursima/article/view/572%0Ahttps://ejournal.indobarunasional.ac.id/index.php/jursima/article/download/572/337>
- [11] Setiaji, A. (2020). RANCANG BANGUN APLIKASI PEMESANAN DESAIN JERSEY BERBASIS ANDROID DENGAN MENGGUNAKAN TEKNOLOGI FIREBASE (Studi Kasus : Konfeksi Minister). *Jurnal Sistem Informasi Dan Sains Teknologi*, 2(2). <https://doi.org/10.31326/sistek.v2i2.664>
- [12] Syamsudin, M. S., Patrio Akbar, A., Ali, A., Rahzisa, D. A., Eliyanti, N. E., Mas Nur'aeni, N., & Purwaningsih, S. (2023). Rancang Bangun Alat Pendeteksi dan Monitoring Banjir Menggunakan ESP32. *Senatib*, 780–787.
- [13] Teknik, F., Bangsa, U. P., Detecting, I., & Natural, F. (2025). *IoT Pendeteksi Bencana Alam Banjir*. 4(1), 191–197.
- [14] Utomo, W. A., Nugroho, I., & Wardhana, O. W. (2022). Kontrol Kecepatan Berbasis Teknologi Wireless Digunakan Pada Kipas Angin Paralel. *Go Infotech: Jurnal Ilmiah STMIK AUB*, 28(2), 121–128. <https://doi.org/10.36309/goi.v28i2.176>
- [15] Windiasmoro, R. D. (2024). Rancang Bangun Prototype Sistem Monitoring Pintu Air Otomatis Pengendali Banjir Berbasis Internet of Things . Rancang Bangun Prototype Sistem Monitoring Pintu Air Otomatis Pengendali Banjir Berbasis Internet of Things Rizki Dwi Windiasmoro Lusia Rakhmawati. *Jurnal Teknik Elektro*, 41–48.